DOCUMENT RESUME

ED 327 384 SE 051 829

AUTHOR Hennessy, Sara; And Others
TITLE Shrink-A-Cube: A Computer Based Activity To Support Primary Arithmetic. CAL Tech. Report No. 73.
INSTITUTION Open Univ., Walton, Bletchley, Bucks (England). Inst. of Educational Technology.
PUB DATE Feb 88
NOTE 26p.
PUB TYPE Guides - Classroom Use - Guides (For Teachers) (052)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS Algorithms; *Arithmetic; Cognitive Development; *Computer Assisted Instruction; Elementary Education; *Elementary School Science; Foreign Countries; Instructional Materials; *Manipulative Materials; Mathematics Education; *Problem Solving; Teaching Methods; Thinking Skills
IDENTIFIERS Apple Macintosh; England

ABSTRACT

This paper describes an educational program developed on the basis that the use of blocks and other manipulatives can be useful both for initial teaching of exchange operations in arithmetic and for remediation of the faulty strategies which very commonly arise during acquisition of written calculation algorithms. This computer program is a multibase blocks microworld adapted for use with the Macintosh Plus microcomputer. The program has a game format involving addition and subtraction operations being carried out with screen representations of concrete objects, namely Dienes blocks. Unlike some other systems, Dienes blocks cannot be broken down into their component parts, but must be exchanged. Blocks in bases other than in 10 are also available in this activity. The aim is to improve children's understanding of place value by encouraging the use of activities based on Dienes blocks alongside the teaching of formal algorithms for arithmetic calculation. Emphasized is the value of peer group teaching or cooperative learning. Included in the document are a discussion of the design of the activity, a "guided tour" of the game, results of field trials, ideas for further development and student modelling, and conclusions. Appended is a sample protocol observed between two students. (KR)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED327384

Centre for Information
Technology in Education

SHRINK- A-CUBE:

A COMPUTER BASED ACTIVITY

TO SUPPORT PRIMARY ARITHMETIC

by

Sara Hennessy, Rick Evertsz, Ann Floyd, Mike Fox, Tim O'Shea

Dave Ellis, Phil Black

CAL TECH. REPORT NO 73
February 1988

SE051 829

Institute of Educational Technology
Open University
Walton Hall
Milton Keynes
Great Britain
MK7 6AA

Shrink-a-Cube: a computer-based activity to support primary arithmetic

by Sara Hennessy, Rick Evertsz, Ann Floyd, Mike Fox, Tim O'Shea, Dave Ellis, Phil Black

(A joint Open University/SYSTEM Ltd. enterprise)

Contents Page

1. Introduction 1
2. Design of the activity 2
3. Playing the game 7
4. Field trials 9
5. Ideas for future development and modelling 11
6. Conclusions 12
7. References 13
8. Appendix: sample protocol 14

1. INTRODUCTION

This paper describes an educational program constructed by the team members of the project, "An Intelligent Computer Tutor for Arithmetic." This three-year project is funded through the UK 'Alvey' programme of advanced information technology research. It involves collaboration between the Institute of Educational Technology and the School of Education at the Open University; the project team consists of Rick Evertsz, Ann Floyd, Mike Fox, Sara Hennessy, Tim O'Shea and Dave Perry. We work in conjunction with a commercial software house (SYSTEM Ltd.) specialising in education and training. The objective of the project is to carry out research and development work on an adaptive computer tutor for arithmetic calculation and for the principles underlying that skill.

Our first program, *Shrink-a-Cube*, is a multibase blocks microworld whose aim is to make explicit the principles of exchange. It is suitable for primary school children aged 6 - 12 and was first developed as a practical activity by the OU course team who produced "Developing Mathematical Thinking" (EM235, 1982). We have adapted the activity for the Macintosh Plus microcomputer. Our program has a game format involving addition and subtraction operations being carried out with screen representations of concrete objects, namely Dienes blocks. The base-10 version of these blocks comprises unit cubes - normally made of wood or plastic - and rods consisting of 10 units stuck together (see Fig. 1 overleaf). Ten of these rods make a 'flat', a square object containing 100 units stuck together, and 10 flats comprise a cube of 1000 units. Unlike some other systems, Dienes blocks cannot be broken down into their component parts, but must be exchanged. Blocks in bases other than 10 are also available in our activity.

Shrink-a-Cube was developed on the basis that the use of blocks and other manipulatives can be useful both for initial teaching of exchange operations in arithmetic and for remediation of the faulty strategies or 'bugs' which very commonly arise during acquisition of written calculation algorithms. It is now widely acknowledged amongst educational theorists and researchers that an explicit link between practical experience and formal representations or symbols is necessary for meaningful learning. However, research indicates that while direct manipulation ought to be a core activity throughout primary schooling, it is not; an APU survey (1985) of the use of number apparatus - including Dienes blocks - showed that 36% of top juniors never used it at all and only 17% used it at least once a fortnight. Unfortunately, such apparatus is often relegated to infant school mathematics and considered to be childish. Our aim is to dispel this view and to improve children's understanding of place value by encouraging the use of activities based on Dienes blocks alongside the teaching of formal algorithms for arithmetic calculation.

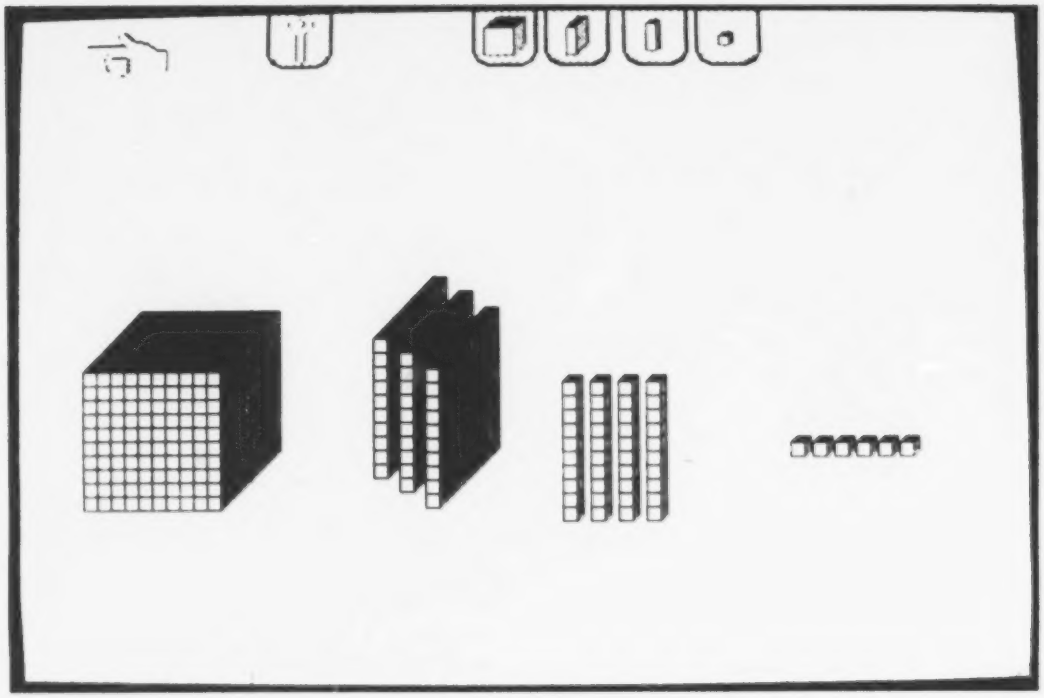

Fig. 1: Dienes blocks display

Finally, while the *Shrink-a-Cube* program may be used by individual children, we believe it is far more appropriate and valuable for a pair or small group of pupils to cooperate in the activity. Social interaction is known to provide the basis for rich learning experiences during activities of this sort. This is because children's discussion of the strategies and key concepts involved helps to build and consolidate their arithmetic representations (Balacheff, 1985; Johnson and Johnson, 1986). One possible explanation for this is that cooperative problem solving (aimed at achieving a joint solution or goal) results in conflict between the participants, and negotiation and compromise are thereby evoked (O'Malley, 1987). In sum, we wish to emphasise the value of peer group teaching using the computer as a tool. It should also be stressed that the teacher plays an important role in our computer-based activity, which is flexible enough to adapt to individual teaching styles; it is not intended to be used independently.

2. DESIGN OF THE ACTIVITY

In this section, we present a detailed description of the *Shrink-a-Cube* interface and discuss both the route to its construction and the ways in which it may be used. The activity revolves around the child's direct manipulation of three-dimensional screen representations of Dienes blocks. For simplicity, the keyboard is not used and all actions are initiated using the mouse. Units, rods, flats (20 of each) and cubes (2) are separately contained within four sliding drawers; they are held off-screen so as to maximise working area. Blocks can be accessed by clicking on a drawer 'handle' which contains an icon indicating its contents. The drawer slides open to reveal its contents (see Fig. 2). The blocks are presented in rows inside each drawer and when a block is returned, it jumps to the space from which it came. Objects are moved about by clicking on them with the cursor - in the shape of a hand - and dragging them across the screen with the mouse button compressed. When the button is released, the objects drop. All objects in the microworld can be physically manipulated in this way. The hand realistically adapts to the shape of each item when picking it up (see Figs. 2, 3, 4 for examples). Objects can be placed anywhere on the screen - blocks need not be ordered or placed in columns - except that blocks of one type cannot be returned to a drawer holding another type. (Children are allowed to create a messy display of blocks so that the reason for systematicity will become apparent to them.) Small objects have priority so that they cannot disappear behind larger ones.

An important control feature is the 'talking duck' which acts as an additional, sophisticated cursor. It can pick up multiple objects at once and move them together so as to avoid laborious sequences of dragging actions. The duck's beak acts as a pointer and is placed against a target object. A mouse click results in the object - however large - being 'swallowed up' and further objects can be assimilated in the same way, providing they are of the same type. With each click, a number appears on the duck's body showing the cumulative total of the objects swallowed (Fig. 2). This number is also spoken when the speech option is selected. The duck is then moved to the target position and a double click results in all of the accumulated objects reappearing on the screen, in a neat row.

Another crucial feature in this microworld is the dice roller, which takes the form of an octopus whose legs hold up two dice with conventional markings. (In an earlier version, a teddy bear was used instead.) When the child clicks on a die, the dice are juggled and a random numerical display then appears. There is a special tool drawer containing the duck and octopus, and a blackboard icon depicting a multiplier (e.g. x 2). This is used to increase complexity of the activity; the child should multiply the total on the dice display by the specified number before proceeding with his/her turn (Fig. 3). A Help option is also available in the tool drawer; it takes the form of a book icon saying 'READ ME' and having an OPEN button. Clicking on this button opens the book to reveal a short - visual and spoken - demonstration; at present it shows 10 unit cubes building up to make a rod.

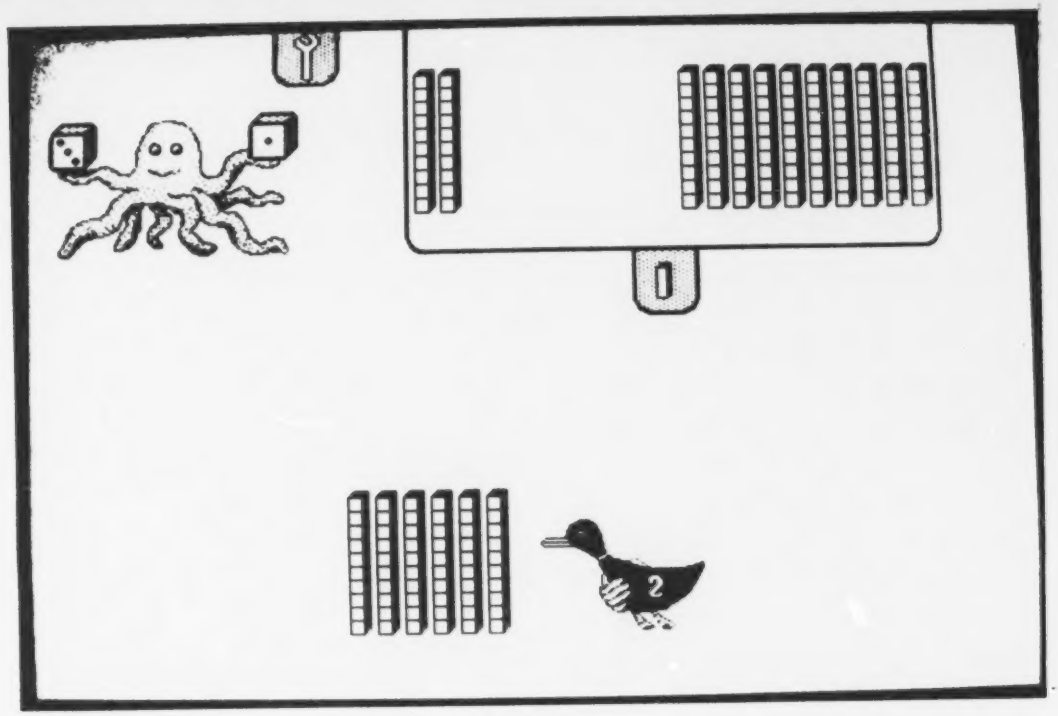

Fig. 2

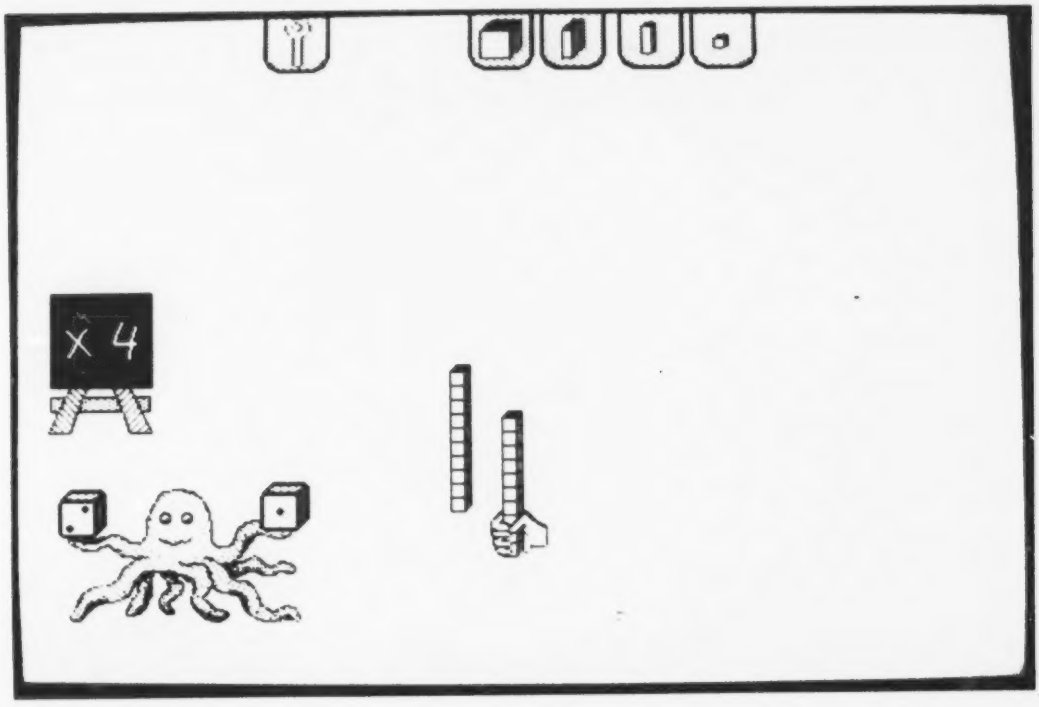

Fig. 3

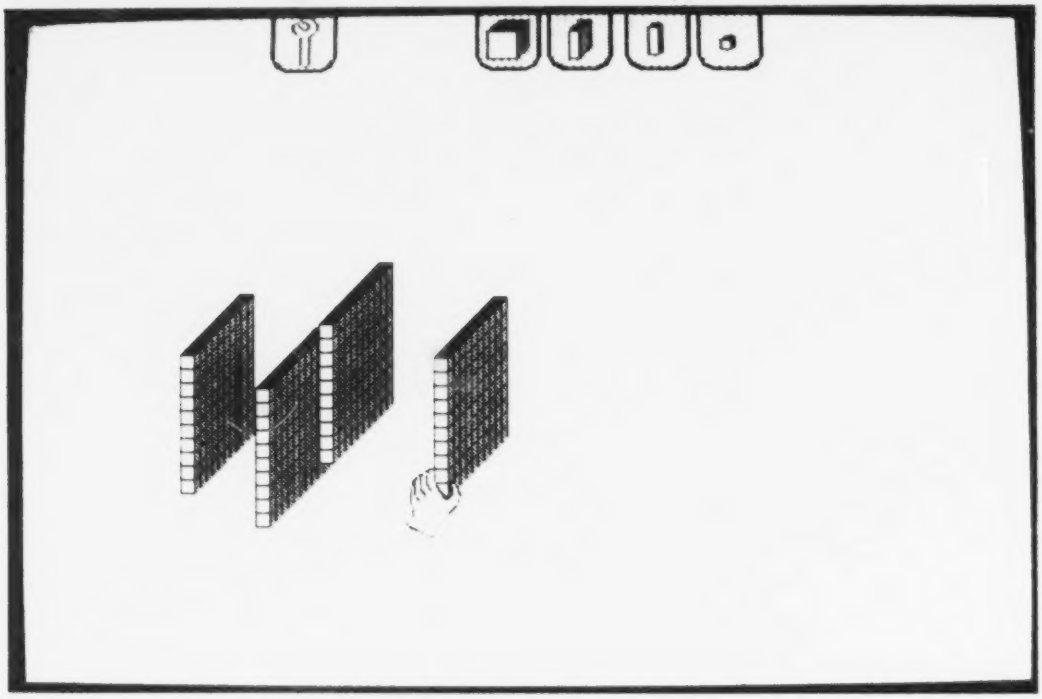

Fig. 4

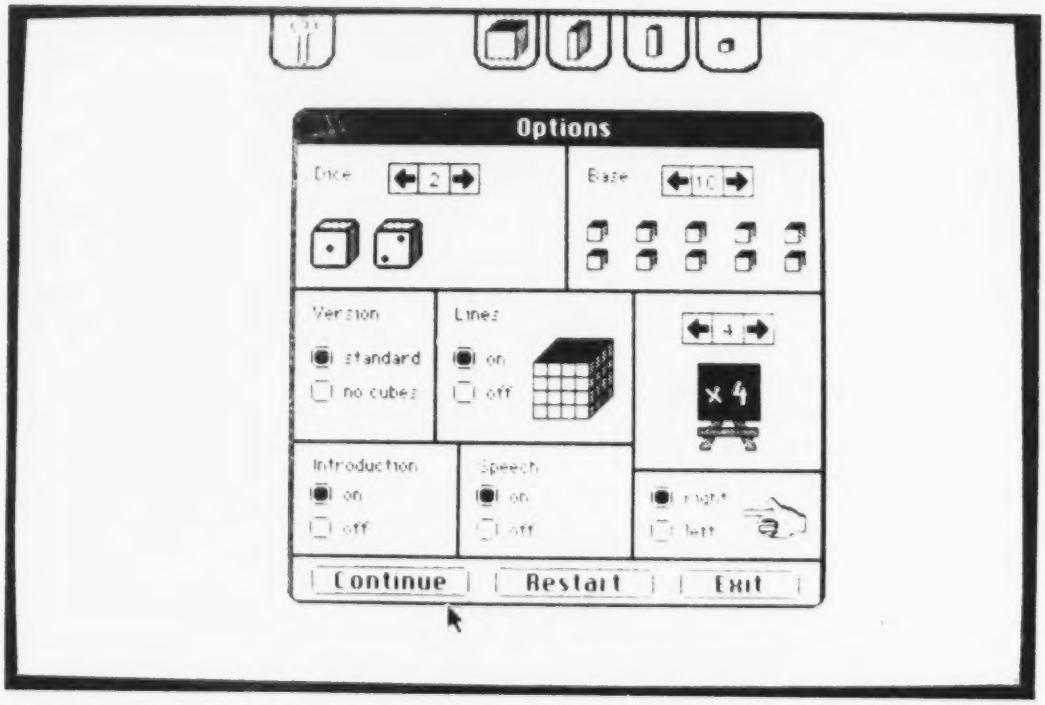
Options
Dice
2
Base
10
Version
standard
no cubes
Lines
on
off
4
x 4
Introduction
on
off
Speech
on
off
right
left
Continue
Restart
Exit

Fig. 5

Finally, there is an options window allowing teachers to manipulate the difficulty level and several other aspects of the activity (Fig. 5). This window is accessed through holding down the options key and clicking the mouse at the same time. It contains options for setting the multiplier (x 2, x 3, x 4 or none at all), for setting the number of dice available (1 or 2), and for changing the number base (from 4 - 10). Altering the base changes the number of divisions on the rods, flats and cubes so that in base 7, for example, a rod equals 7 units. The options window also provides facilities for choosing a lefthanded versus righthanded cursor (see Fig. 4), removing all line divisions on the blocks, turning speech on/off, turning the Introduction (described below) on/off and choosing between the standard or a 'no cubes' version of the activity. RESTART and EXIT buttons are provided too.

Technical Description of the Interface Design

The *Shrink-a-Cube* activity required good high-resolution graphics, smooth animation and a mechanism for the user to manipulate objects on the screen with ease. Since there could be many objects on the screen at one time (up to 90) with very few restrictions on their positioning, it was necessary to design a flexible multisprite animation subsystem on which to base the activity. To obtain smooth animation, two screens are required. The program always draws on the 'hidden' screen and once finished flips this into view. This avoids any flashing on screen. The drawing routines themselves needed to be very fast, and because large numbers of sprites were involved, the animation routine had to be designed so that it did not slow down as more and more sprites were displayed on screen. The routine was required to update the screen every 50th second (one frame period). Other requirements were (a) for the sprites to be designed in the MacPaint package and transferred to the program; (b) for flexible control over the sprites in terms of setting their destination, velocity, acceleration or defined path function; (c) for an ordered priority of sprites on screen and for animation sequences made up of predetermined frames. Later requirements were for a cursor defined as a sprite, and (d) for sprites to be linked together for easier control.

From the outset is was apparent that the Quickdraw routines available in the Macintosh operating system would not be either fast or flexible enough to allow any reasonable animation. The standard facilities are fine for writing word processors and spreadsheets but are completely inadequate for *Shrink-a-Cube*. Ultimately the only operating system routines to be used at all in the main activity were for loading and saving a file in bytes and for the speech output. Additional routines were used to implement the teachers' options window. The vast majority of the code was written in the C programming language for speed of execution and portability to other microcomputers. Small, time-critical sections of code were written in assembler to operate at the maximum possible speed; this was essential for fast, smooth animation.

The sprite animation system was based on a virtual screen of 64k x 64k co-ordinates, each pixel being 32 x 32 virtual screen co-ordinates. This provided very accurate plotting of sprites, giving smooth sprite trajectories. The actual Macintosh screen is 512 x 342 pixels, which is about 1/20th of the virtual screen area. There is thus plenty of space off screen (not visible) to which sprites can move. In fact, in *Shrink-a-Cube,* there are about 250 sprites in all, with at least 160 of them 'drawn' off screen. In reality, of course, the animation routine does not actually attempt to draw them.

The sprites themselves are held in two doubly-linked lists, one for all sprites and one for the sprites currently on screen. This provides the order for plotting overlapping sprites. Sprites may be inserted or deleted from these lists. Because it would take far too long to draw all the sprites that could possibly be on screen during every frame period (1/50 sec), the routine was designed only to draw those portions of the screen that had actually changed from the previous frame.

The most innovative piece of design enabled recognition of which sprite the user was pointing at with the mouse, even though many sprites could be overlapping each other. The routine was even capable of selecting one sprite through a hole in the sprite covering it! Other innovations involved detection of sprite collisions, automatic sprite trajectories, the automatic spacing of sprites dropped in a pile on screen (e.g. to avoid losing a sprite) and a different cursor (or hand) for each type of object. The latter gave the impression that the hand was realistically grabbing or lifting the block or rod, or that the duck was dragging open the drawers with its beak.

3. PLAYING THE GAME

A short guided tour of the game is provided first. At present, this is a demonstration showing 10 unit cubes building up to make a rod, 10 rods building up to make a flat, and 10 flats building up to make a cube. As the blocks slide into place (from off-screen), they are counted aloud and as each object is completed, the computer voice tells the user "10 rods make one flat..." If we had developd the acitivity further, this introduction would eventually have included use of the mouse, object manipulation and a demonstration of the various microworld control features. If the introduction is turned off in the options window, the computer voice simply says 'Hello' at the beginning of the game.

The easiest version of the game to play is *Build-a-Cube*, whose objective is to build up blocks to reach a predetermined goal such as a cube of 1000 units. In order to understand the more complex game *Shrink-a-Cube*, this version ought to be played first. Shorter versions of the activity include *Build-a-Square* or even *Build-Two-Rods* for very young children. Each version concentrates on fostering addition skills as well as developing children's understanding of place value. Teachers or the children themselves may set the goal. Although the activity is suitable for individual children, it is probably more valuable and more interesting when played by a pair of children and will thus be described that way

here.

In *Build-a-Cube*, the children take turns throwing the dice, adding up the numbers of dots in the dice display, and removing from the drawers the total number of units shown in the display. If the multiplier is used, they must multiply the sum on the dice by the appropriate amount in order to obtain the total number of units to be removed. The blocks removed from the drawers are placed in a pile designated to the active child, so that two separate piles are built up (one in front of each child: see Fig. 6). The children are expected to perform exchanges; indeed, they will very quickly run out of units in the drawer and will have to exchange 10 of their accumulated units for a rod. One child may have to ask the other to perform an exchange so that s/he can proceed, and this usually provokes a good deal of discussion between the pair. Other children may join in the game by verbally instructing the players in making exchanges. The children proceed until one of them wins by achieving the desired goal. *Shrink-a-Cube* is very similar except that the children begin with a particular display, e.g. a cube, and must 'shrink' it down to nothing by taking turns in removing the number of units shown on the dice. This version provides practice in both addition and subtraction skills.

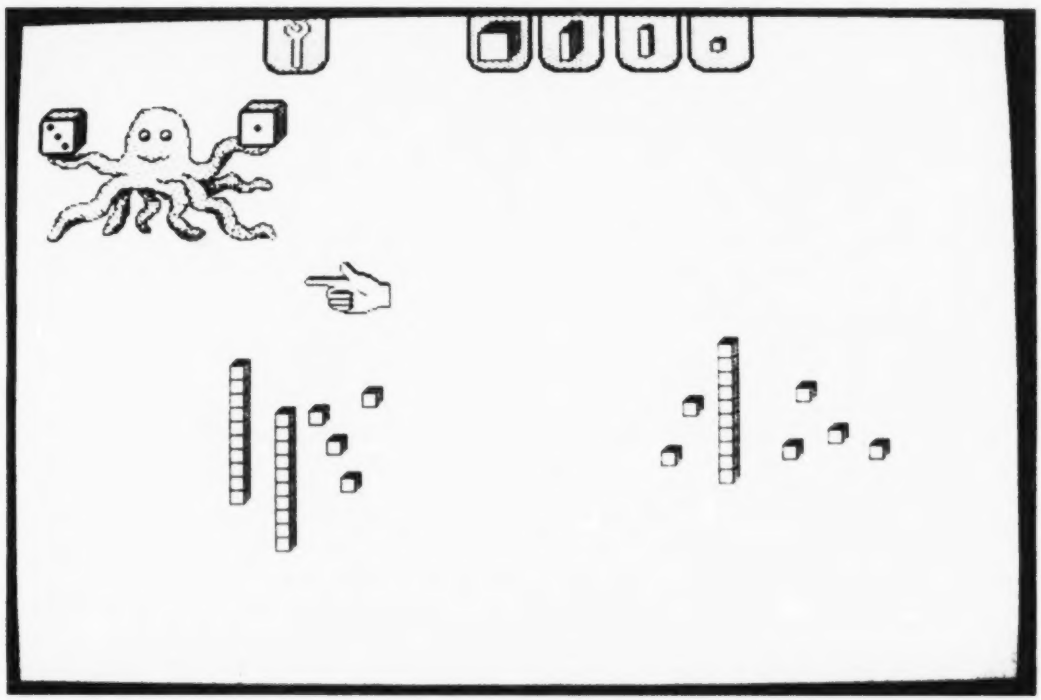

Fig. 6

4. FIELD TRIALS

A series of field trials was carried out with 35 children in local primary schools. The purpose was to assess certain features of the interface and to investigate children's ability to grasp the point of the game. We were also interested in whether children playing in pairs would compete with or help each other, show more competence or grasp the game more quickly than children playing alone. Seven individuals and five pairs of children aged 6 - 12 were tested in a Milton Keynes middle school by the OU project team. A further six individuals and six pairs aged 7 - 11 were tested in three Sheffield schools by the SYSTEM team. Sessions typically lasted about 30 minutes; they were carried out in an empty classroom and were recorded on audio tape. The children first viewed the guided tour and then the experimenter demonstrated mouse control and the action of each screen object. Each child was given a few minutes practice in moving objects about on the screen, opening drawers and so on before the experimenter explained the rules of the game and set the goal (e.g. three rods). The children then played *Shrink-a-Cube* or *Build-a-Cube* or both (depending on age, ability and time constraints) under close supervision of the experimenter. S/he intervened whenever necessary, giving help with object manipulation and asking probing questions when a child became stuck. The nature of these interventions is illustrated in the sample protocol in the Appendix on page 14. Note that there was no competitive element for children playing alone; they simply tried to reach their goal.

We found that children of all ages pick up mouse manipulation easily and rapidly - more so than some adults - so that this mode of input did not prove to be a problem. Nevertheless, a further experiment was carried out with seven individual children in order to assess whether a Mactablet and pen were easier to use than the mouse. The former input device proved to be no more popular than the mouse and was thus abandoned. Interestingly, several children mentioned that mouse control was hard to master, despite showing marked proficiency in this area.

Results

Interface design

Most of the children tested had had some prior experience with Dienes blocks in the classroom, and no difficulty in identifying these and other objects on the screen was observed. Children accepted the reality of the microworld without question. The most popular control features were the teddy bear and octopus dice rollers and the 'magic' duck; contrary to our prior fears, these features did not prove to distract children from the game. In fact, children occasionally forgot to use the duck as a shortcut when picking up large numbers of objects. There was a lot of interest in how objects re-order themselves. Children of all ages and both sexes portrayed much enthusiasm for the activity, yet this may well be related to the novelty effect of engaging in a computer-based activity containing interesting animals and other such features. Most of the children interviewed had indeed had limited, if

any, prior computing experience.

Advantages of *Shrink-a-Cube*

The cooperative problem-solving environment appeared to enhance the children's experience greatly and much more progress was observed amongst those working in pairs. Those children tended spontaneously to instruct, help or criticise each other, rather than to compete. For instance, it was often a child's partner who suggested that the child trade in excess units for a rod when none remained in the units drawer. Of course, not all of the suggestions made were practical. There were occasional instances of dominance by one partner and conversely, one child persisted in ignoring his partner's helpful remarks. A few children were found not to work well with others; discussion was promoted only when the experimenter intervened. Overall, however, this mode of working seemed to be very successful. It proved easier for children to grasp use of the interface tools when working cooperatively because they tended to explore those features together.

Another advantage of the activity is that it is directly analogous with manipulation of real Dienes block materials, and these were known to be familiar to our sample. However, further information about transfer between work with the concrete materials and manipulation of the more abstract screen objects is lacking. It was beyond the scope of our project to investigate this issue in depth. More general information on the value of the activity to pupils can be found in the companion technical report by O'Shea (in preparation).

Disadvantages of *Shrink-a-Cube*

An obvious drawback of the program at its current stage of development is that discussion of the skills and concepts involved in playing the game requires an enormous amount of mediation by an experimenter or teacher. Secondly, no need to conserve quantity was imposed in the activity, and the point of exchange transactions appeared to be unclear to some children who did not spontaneously adhere to the principles of exchange. This would have led to cheating in the absence of an experimenter, and more importantly, it may hinder learning.

The activity does not easily permit mental arithmetic; its aim is not clear in this context as it primarily supports written calculation (or acts as its precursor). Rather than dealing with place value, it emphasises the principles of exchange. Representing the standard 'fused' Dienes blocks means that regrouping 100 to get ones involves two trading procedures, and that process does not parallel the steps of the regrouping process as applied to number symbols. It also clouds the issue of whether the number is preserved in regrouping. One means of overcoming this was proposed by Champagne and Rogalska-Saz (1984) from LRDC. They have developd a program based around '100 cubes,' each containing 10 long boxes of 10 unit cubes. These allow regrouping to be performed in one

step. Since our fieldwork indicated that several children in fact wanted to break up rods physically into units, some consideration ought to be given to this idea.

Although it was decided not to develop the program any further within the course of the present project, some ideas for future improvements have been formulated and they are now presented.

5. IDEAS FOR FUTHER DEVELOPMENT AND STUDENT MODELLING

Interface development

Some interface features which could be implemented include the following: a 'Ready Teddy' figure for signifying the end of a turn, a PARDON button for children's use when they wanted an intervention to be repeated, a vacuum cleaner to replace the magic duck - for swallowing up large numbers of objects more realistically, and a replay facility so that children could monitor their partners' turns or check their own working (and teachers could monitor children's progress). An odometer or measuring device could also be introduced. This might use a number line analogy to provide a record of progress and of distance from a visible goal (e.g. one flat) at any required point in time.

Another important feature is a divided screen with a zoom facility so that two players would have separate, labelled working areas and their displays could not be confused or combined. Children would need access to both play areas so that exchanges could be made using an opponent's blocks when not enough blocks remained in the drawers. Similarly, both players' block displays must remain visible at all times, even if one area is permitted to shrink in size. Finally, it would probably be useful to create another, simpler version of *Shrink-a-Cube* which contains large-size rods and units (only) for use by younger children.

Diagnosis and intervention

It is clear that a number of diagnostic and feedback mechanisms will need to be built into the *Shrink-a-Cube* microworld if it is to be divorced from constant experimenter/ teacher intervention and if we are to measure how much learning has taken place. Construction of those mechanisms would involve a very detailed content analysis of the task in terms of the arithmetic skills and concepts underlying it. Then, a student model might be built up through tracking children's operations during a game, asking them simple Yes/No questions, and presenting a range of specific, related tasks so as to try and divine what is not understood. (Dice displays could even be predetermined or children could be asked to solve a problem from a particular point.) Since we know that children do not use blocks in the classroom as often as they usefully could, we might have to begin with very simple

tasks such as getting children to represent different numbers by removing appropriate objects from drawers, and asking them to 'read' block displays of various magnitudes.

A major problem at present is that there are no constraints on the validity of exchanges. Children are allowed to make errors such as removing the wrong number of units from a drawer in exchange for a rod. Eventually, the computer would need to be able to recognise and comment upon incorrect actions. In order to achieve these checking and advisory functions, the program would need to know not only that a mismatch between, say, 9 and 10 units has occurred, but also what the children are trying to do. In other words, it must know the children's objectives and be able to keep track of events without making the activity too restrictive. Whilst some strategies are more efficient than others, various alternatives can result in a correct action. We ought to let children use their preferred strategies at first - so that they build up confidence - and later encourage them to seek alternatives. One useful device might be the introduction of a simulated child who makes occasional mistakes, or who uses very sophisticated strategies by way of example. Ideally, microworld activities should consist of a set of goals with various methods of achieving them, but we also want children to be able to invent new methods. Our dream for the future is that children should actually be able to teach new games to the computer!

6. CONCLUSIONS

In this paper, we have presented an account of our development of *Shrink-a-Cube*, a Dienes blocks microworld, on the Macintosh computer. A description of both the technical and educational design work has been outlined. The aim of the activity is to foster arithmetic understanding by making explicit to primary school children the principles of exchange. We have reported a series of field trials which investigated children's ability to grasp those principles. In particular, we assessed the value of a joint problem-solving environment and found that working in pairs was on the whole very successful. We believe that this kind of environment evokes greater cooperation and invention of imaginative strategies, and that it facilitates exploration of the program's control features. Of course, children need to check the validity of strategies suggested by their partners. In sum, we should encourage children to work together and to learn from others' input but not to accept blindly the suggestions made by other children. A technical report giving more information on pupil use of *Shrink-a-Cube* is forthcoming (O'Shea, in preparation).

Shrink-a-Cube is an immensely flexible activity and there are currently many different ways of using the environment. There is thus room for development of a range of activities which might be set within the microworld, as well as for further improvements and the implementation of various modelling strategies (some of which have been outlined above) pertaining to the *Shrink-a-Cube* activity. The program will not be refined any further by our project team, but its development and testing have yielded some useful design guidelines for future educational activities of a similar kind. For instance, it is clear that a game's goal, rules and imposed constraints must be made very clear and must appear

sensible to the child, so as to avoid the child's circumvention of them. The use of familiar objects such as multibase blocks is recommended. *Shrink-a-Cube* has additionally served as an ideal vehicle for the development of innovative interface designs, a key one being the sprite animation system.

7. REFERENCES

Assessment of Performance Unit (1985). A Review of Monitoring in Mathematics: 1978 to 1982. London: HMSO.

Balacheff, N. (1985). Experimental study of pupils' treatment of refutations in a geometrical context. Proceedings of the Ninth International Conference for the Psychology of Mathematics Education, Noordwijkerhout, Holland.

Champagne, A.B. and Rogalska-Saz, J. (1984). Computer-based numeration instruction. In V.P. Hansen (Ed.), Computers in Mathematics Education. 1984 Yearbook: National Council of Teachers of Mathematics.

Hennessy, S., Evertsz, R., Ellis, D., Black, P., O'Shea, T. and Floyd, A. (1988). Design specification for 'Shopping on Mars', a computer-based educational activity. Centre for Information Technology in Education technical report no. 29, Open University.

Johnson, D.W. and Johnson, R.T. (1986). Computer-assisted cooperative learning. Educational Technology, January, 12-18.

O'Malley, C. (1987). Explanation based learning by computer users. Paper presented at the Workshop on Complex Learning, organised by the University of Lancaster Centre for Research on Computers and Learning. Grange-over-Sands, Cumbria.

O'Shea, T. (in preparation). Pupil use of Shrink-a-Cube. Computer-Aided Learning Research Group technical report, Open University.

8. APPENDIX: Sample Protocol of Joanne (J) and Teresa (T) Playing *Shrink-a-Cube*, with Sara Hennessy as Experimenter (E)

Build-a-cube

(J throws dice.)

E What have you got?

J Nine. (J removes 9 units)

E Okay, will you have a go?

(Long pause. T throws dice, gets 2 and removes 2 units.)

E You have to go a bit nearer, put the beak right on it. Good, okay, you have a go now.

(J throws 10 and removes a rod.)

E That was a nice easy one, wasn't it? (T throws dice.) Now what have you got to get?

T Nine. (T removes 9 units.)

(Long pause. J throws a 2.)

E Just pick it up, that's it. (J looks horrified to find no units left.) Oh dear, what are we going to do?

J Are you allowed to take any of those?

E Any of those what?

J Those boxes.

E The rods you mean, the big ones?

J Yes.

E Well you can take anything out you like as long as you end up with two there. How can you do it?

(Long pause)

J I don't know.

E Well how many have you got here?

J Nineteen.

E Nineteen and you want to add two, don't you? So how many would you have when you've done that?

J Twenty-one.

E Now is there any way you can think of to get twenty-one there?

J Put them back and take one out of there and get two tens.

E That's a good idea.

(J returns her 9 units and then removes 1 unit and a rod.)

E Do it twice, that's it (meaning double-click to release blocks from duck).

(Long pause)

E Good, is that right, what she's done?

T Yes. (T throws dice and gets 3.)

E You have to move the duck.

T How do you do that?

E Well, put the hand back on the duck.

T Oh, I don't know what I'm doing.

E Well, you've got to get the tip of the finger onto the duck... move the hand over a bit, that's it, good.

T Shall I put it back on?

E Yes.

(Long pause. T removes 3 units. J throws an 8 but only 5 units remain in drawer.)

E There's not enough, are there?

T She could take those (rods) and take those big boxes.

E Well, there would be more than eight then, wouldn't there, because remember the rods are worth ten and the bigger ones are worth even more than that, and she only needs eight.

J Eight and one is nine, nine from there.

E There's not enough there for you either.

T Well, she could throw the dice again.

E Oh no that would be cheating, she's got to do eight. Is there any way you could help?

T Could be, I could give her a few of mine.

E Well that wouldn't be fair though would it? Is there another way you can think of? Is there anything you can think of, Joanne, that she might help you with? What about the blocks that Teresa has? She has got most of the units on her side, hasn't she, that doesn't seem fair.

J I can swop some with her with a ten.

E I think that's a very good idea.

T If there's nine there, you could put one of them and get ten back and get a ten.

E Do you want to do that, that would help wouldn't it, putting some more back in the drawer. So what do you need to do then?

J Put ten back.

(Long pause)

J Oh.

E I think it swallowed one when you weren't looking. That's it, you've still got the same amount now, haven't you, so that's fair.

(Long pause. J removes 8 units and one of them partially covers another.)

E It's alright, you can still just about see it can't you? Good. Now it's your turn.

(T throws a 7.)

T Can I touch that over there?

E Yes, they both work. You might have to pick up the mouse though because it's going off the edge, that's it. You are lucky, there's just seven left.

(T removes 7 units. J throws dice and gets 9.)

J Ah.

E What's the matter?

J It's made one of those things.

E None left, oh dear, what can you do?

J I can do that swopover because there's a ten there.

T Shall I get a ten out of mine and put these ten back?

E You could do.

(Long pause)

E That's it, now you've got most of the units haven't you?

T Pardon?

E I said now you've got most of them on your side.

J Nine, I can do mine and swop them for the ten and then...

E Yes, you've got a lot this time haven't you?

T Although there's nine there. Could she leave those there and get a ten?

E She could do, she could have ...

J Ten, so I've got one in the drawer.

E That's alright, you can open it with the duck full.

(Long pause)

J Oh, oh no!

E Did you take one too many? Why don't you put those down and then put them back.

T Take one back.

E That was quite a lot of swallowing, wasn't it.

(T throws dice and gets 6.)

J Oh no. I've got ten left, I need to change.....

E Well hang on now, let's see if Teresa can do it just using her own this time. We could do a swop with yours but that would make it easy wouldn't it? Let's see if there is a way she can think of. Now there's not enough in there, are there, and you need to get six. How many have you got in your display altogether?

T Five.

E Five units and how many others?

T And two tens.

E Yes.

T Oh, can I put a ten back...

E You can do something like that... what could you do? You need to get six out, remember, and you've got two tens and a five.

T Can I put a ten back and pick up about three or ... out of ten?

E I think so, I'm not sure what you mean, try it.

(Long pause)

T Did I sit on him or did it gobble it up?

E No you ate one up from here.

T Can I take three of them from the ten?

E Three units from the rod do you mean?

T Yes.

E Well you can't break these ones, they're solid. So you'll have to think of another way, another combination.

T I can't think of anything.

E Alright, how many have you got here altogether?

T Twenty-five.

E Right, and you want to add six, don't you?

T Yes.

E So if you added six, what would you have... any idea?

T Thirty.

E Nearly!

T Thirty-one.

E That's right, that's what you need to end up with, isn't it? So what could you do to get 31 in your display?

T Oh, I know... (T finally constructs the most efficient strategy possible, returning 4 units to the drawer and removing a rod.

E Very good, you've done it, haven't you, you've ended up with thirty-one. Did you understand what she did, Joanne? Do you think that was right? (J nods.) Right, I think we'll stop there because otherwise we could be building up for ever and ever. Now we'll play 'Shrink-2-rods' with Joanne to start. There you go.

Shrink-a-cube

(J throws 3 on the dice.)

E Now you've got to take three away.

J I'm trying to break them off.

T Can we break the rods?

E I don't think they'll break.

J He'll eat it up.

E You'd need a hatchet. Yes, he'll eat it up if you try and break them. How could you put three away in the drawers?

J I don't know.

E Well, think about what we did in the last game because this is quite similar, this one.

J & T (simultaneously) Oh... can we swop the ten with the units?

E That's right.

J Shall I just get seven because... (once again, J has spontaneously suggested a more efficient strategy.)

E That would be quicker, wouldn't it, good. Now how can you take away eight? Very good, okay, we'll stop there for today.

List of Technical Reports

These reports may be obtained from:-

Hansa Solanki, Institute of Educational Technology, The Open University,
Walton Hall, MILTON KEYNES, MK7 6AA, England.

Report No.	Title and Author
1	M.A. Bramer (1980) Using Computers in Distance Education: The first ten years of the British Open University.
3	M.A. Bramer (1980) COMAL-80 - Adding Structure to BASIC.
4	A. Jones (1980) An evaluation of the SOLO programming language: A pilot study of its use at summer school.
5	A. Jones, T. O'Shea (1981) An evaluation of tutorial CAL at the Open University: the use of Cicero in SDT286.
6	E.L. Whitelegg (1981) An evaluation of tutorial CAL at the Open University: the use of MERLIN in ST291.
7	M.W. Lewis (1980) Improving SOLO's User-Interface: An Empirical study of user behaviour and a proposal for cost-effective enhancements to SOLO.
9	T. O'Shea A. Floyd (1981) Recording Children's Mathematical Behaviour.
10	A.T. Vincent (1981) Computer-assisted support for blind students: the use of a microcomputer-linked voice synthesizer.
11	P. Bacsich (1981) Viewdata-style delivery mechanisms for CAL.
12	P.J. Murphy (1981) Biological simulations in distance learning.
13	P.J. Murphy (1981) Computer simulation in Education: Genetics and Evolution.
14	I. Every (1981) Graphics and animation in teaching dialogues.
15	P.J. Murphy, E. Scanlon, E.L. Whitelegg (1981) An evaluation of simulation CAL in the Science Faculty during 1980.
16	M, Eisenstadt, J. Laubsch, J.H. Kahney (1981) Creating pleasant programming environments for cognitive science students.
17	J. Laubsch, M. Eisenstadt (1981) Domain specific debugging aids for novice programmers.
18	J. Roberts (1981) Are fuzzy grids really necessary?
19	J. Preece (1981) Investigating how students interpret cartesian graphs.

20 E. Scanlon, Tom Vincent (1981) CAL in Science: Some Regional Implications.

21 A. Jones (1981) The representation of conceptual models: some issues.

22 E. Scanlon (1982) Improving Problem Solving in Physics.

23 M. Sharples (1982) CYCLOPS - A case study in the design of user-friendly educational technology.

24 T. Hasemer (1982) Mac SOLO.

25 A. Jones (1981) How do novices learn programming?

26 A.G. Priest (1981) A design for an Intelligent Mechanics Tutor.

27 A. Jones, and E. Scanlon; (Eds) A review of research in the CAL group: A report of the first annual conference, November 1981.

28 R. Evertz (1982) A production systems account of children's errors in fraction substraction.

29 A. Jones (1982) Mental models of a first programming language.

30 P. Smith, P.I. Zorkoczy (1982) RADIOTEXT.

31 P.J. Murphy (1982) Interactive Computer Tutorials in Open University Biology Courses.

32 A. Jones, and E. Scanlon (1983) Protocols and production systems.

33 J. Preece (1983) A study of pupils' graph concepts using an interactive computer program.

34 J. Preece (1983) Graph interpretation and sketching skills in 14 and 15 year olds.

35 J. Preece (1983) Exploring pupils' strategies for interpreting qualitative multiple curve cartesian graphs.

36 E. Scanlon, C. Hawkridge, T. O'Shea (1983) Modelling Problem Solving: scripts and production rules.

37 E. Scanlon, C. Hawkridge (1983): Data from two schools.

38 I. Every, and E. Scanlon; (1983) An Evaluation of the CAL at the Summer School for the Course S271 - Discovering Physics.

39 E. Scanlon, C. Hawkridge, R. Evertz; (1983) Modelling Physics Problem Solving: Production Rule Models of Two Problems.

40 Jackson, P. and Lefrere, P. (1983) On the application of rule based techniques to the design of advice giving systems.

41 A. Jones (1984) Learning to program: Some protocol data.

42 Blenkhorn, P.L. and Every, I.M. (1983) A Template Editor for the production of CAL Tutorials.

43 Every, I. and Scanlon, E. (1984) Discovering Physics with Microcomputers.

44 Preece, J. and Jones, A. (1984) The Educational Software Pack for the Open University Project, Micros in Schools.

45 Preece, J. and Jones, A. (1984) Training teachers to select educational computer software: results of a formative evaluation of an Open University pack.

46 Edwards, A.D.N. (1984) Prospects for the Use of Computer Aided Learning in Special Education

47 O'Shea, T. (1984) The Open University Micros in Schools Project.

48 Jones, A. (1984) How novices learn to program: Some protocol data of students learning assembly

49 Surgey, P. and Scrimshaw P., (1985) Who controls CAL? The case of TRAY

50 Preece, J. (1985) Pupils' concepts of gradient.

51 Preece, J. (1985) Some theoretical issues related to interpreting graphs.

52 Vincent, A.T., (1985) Computing and the Blind"

53 Edwards, A.D.N. and O'Shea (1986) Making graphics-based programming systems usable by blind people.

54 Holland, S (1986) How computers are used in the teaching of music and speculations about how artificial intelligence could be applied to radically improve the learning of compositions skills.

55 Scrimshaw, P (1986) What can be learned from TRAY: practioners' perceptions

56 Scrimshaw, P (1986) Teaching TRAY to Teacher

57 Scrimshaw, P (1986) Matching tasks to learners: a preliminary analysis

58 Swann, P (1986) Computer Assisted Grammar of English (CAGE) for Italians: First Report on an Experiment Conducted at the "Blaise Pascal" Secondary School in Reggio Emilia, Italy from 1984 to 1986

59 Baker, M (1987) Intelligent Computer-Aided Instruction and Musical Performance Skills.

60 Edwards, A (1987) Evaluation of *Soundtrack*, a word processor for blind students.

61 Baker, M (1987) Proposed Research Directions for Intelligent Computer-aided Instruction in Musical Performance Skills.

62 Scanlon, E (1987) and Smith, R.B., A Rational Reconstruction of a Bubble Chamber Simulation Using The Alternate Reality Kit.

63 Elsom-Cook, M, (1987) Intelligent Computer-Aided Instruction research at the Open University.

64 Edwards, A., (1987) Integrating Synthetic Speech With Other Auditory Cues in Graphical Computer Programs For Blind Users.

65 Holland, S., (June 1987) A knowledge-based tutor for music composition.

66 Holland, S., (June 1987) New Cognitive Theories of Harmony Applied to Direct Manipulation Tools for Novices.

67 Fung, P., (November 1987) Novice Prolog Programmers.

68 Priest, A.G., (November 1987), Modelling Student Errors in Physics Problem-Solving.

69 Fung, P., DuBoulay,B., & Elsom-Cook, E., (November 1987) An initial taxonomy of novices' misconceptions of the Prolog interpreter.

70 Scrimshaw, P.,

71 Scrimshaw, P.,

72 Eileen Scanlon, Tim O'Shea (February 1988) Learning Issues in Intelligent Tutoring Systems

73 S. Hennessy, R. Evertsz, A. Floyd, M. Fox, T. O'Shea, D. Ellis, P. Black, (February 1988) Shrink-a-Cube: a computer-based activity to support primary arithmetic.